Análisis de Seguridad y Mitigación en un Módulo de Administración

A continuación, se presenta la identificación, la técnica de mitigación y la justificación para cada uno de los cuatro problemas de seguridad detectados.

Caso	Vulnerabilidad Identificada	Técnica de Mitigación	Justificación
1	Cross-Site Scripting (XSS)	Utilizar la **validación de entrada y el escape de salida**. Esto implica filtrar los datos que ingresan los usuarios en el servidor y "escapar" los caracteres especiales (<, >, ", ') antes de que el contenido se muestre en la página HTML.	Al escapar los caracteres, el navegador los interpreta como texto en lugar de código ejecutable, lo que previene que un atacante inyecte *scripts* maliciosos para robar información de las sesiones de otros usuarios.
2	Fallo en el Control de Acceso	Implementar un **Control de Acceso Basado en Roles (RBAC)** en el *backend*. Esto garantiza que el *endpoint* solo sea accesible por usuarios con los permisos correctos y que estén debidamente autenticados.	Esta técnica reduce significativamente el riesgo de una fuga de datos masiva, ya que asegura que solo el personal autorizado pueda acceder a la información confidencial de los usuarios.
3	Uso de Componentes con Vulnerabilidades Conocidas	**Actualizar la librería** a una versión que corrija la vulnerabilidad crítica. Se puede usar una herramienta de análisis de composición de *software* (SCA) como Snyk para escanear y monitorear las dependencias.	Mantener las dependencias actualizadas es una práctica fundamental de seguridad que elimina vulnerabilidades conocidas públicamente, lo que reduce drásticamente la superficie de ataque y el riesgo de que un atacante explote un fallo para comprometer el sistema.
4	Falsificación de Petición en Sitios Cruzados (CSRF)	Implementar **tokens anti-CSRF** únicos para cada sesión.	Este método asegura que cada solicitud para una acción crítica, como "Eliminar usuario", provenga directamente de una interacción legítima del usuario en la página, y no de un *script* malicioso en un dominio externo.

2. Prioridad de Corrección y Justificación

El problema que se debe priorizar para corregir de manera inmediata es el

acceso no autorizado al *endpoint* /admin/users/export (Caso 2).

Justificación:

- **Impacto Crítico:** La vulnerabilidad permite que cualquiera con la URL del *endpoint* acceda a los datos de todos los usuarios sin necesidad de autenticación. Esto representa un riesgo de fuga de información masiva, lo que podría tener graves consecuencias para la empresa.
- **Facilidad de Explotación:** El ataque es trivial. No se requiere hacking complejo, solo una simple petición GET a una URL pública, lo que lo hace el riesgo más urgente y peligroso de los cuatro.